IN THE SPECIFICATION:

Please amend Paragraph [31] as follows:

[31] FIGURES 4a-4h show ~~FIGURE 4 is~~ an exploded version of the data flow diagram of FIGURE 3, showing an algorithm used for reading memory configuration in a preferred embodiment;

Please amend Paragraph [96] as follows:

[96] In the former case (step 204), the generator performs the required action (step 205) and then either stops execution or goes on to the next argument, depending on the actual parameter; in the latter case (206), the generator stores data in a corresponding global variable (step 207), and then checks (step 208) whether further arguments are available. If so, the algorithm loops back to step 203 to read the next argument, otherwise it performs a check (step 209) on the current value of global variables, and sets default values in the case that some of these variables have not been initialized. The process then returns to the main function in step 210.

Please amend Paragraph [98] as follows:

[98] A data flow diagram exploded from the data flow diagram of FIGURE 3 in conjunction with the list of tokens above is given as reference in FIGURES 4a-4h ~~FIGURE 4~~.

Please amend Paragraph [213] as follows:

[213] Data concerning each port is stored at level 2, wherein each port block includes one or more cycle blocks and one behavior block, as shown in FIGURES 4a-4h ~~FIGURE 4~~.

Please amend Paragraph [233] as follows:

[233] In the exploded data flow diagram of FIGURE 9, which relates to a preferred embodiment, each of steps 352, 354, 356, 358, 360,

364 and 366 ~~to 359~~ refers to a specific test, which is generated to a corresponding algorithm 353, 355, 357, 359, 361, 363, and 300 ~~360 to 367~~.

Please amend Paragraph [234] as follows:

234 Test 352 and Algorithm 353 ~~360~~ tests for corrupt entire memory, and generates Verilog code to force a read cycle on the port under test. If the data at all locations equals unknown ("X" value), then it is assumed that the entire memory has been corrupted. Otherwise, a statement to flash a message in a report file is written.

Please amend Paragraph [235] as follows:

[235] Test 354 and Algorithm 355 ~~361~~ tests for corrupt latched location (M-4.2) and generates Verilog code to force a read cycle on the address location that can be corrupted due to a violation. If the data at that location (address) equals unknown, then it is assumed that the latched location has been corrupted, else a statement to flash a message in the report file is written.

Please amend Paragraph [236] as follows:

[236] Test 356 and Algorithm 357 ~~362~~ tests for memory no change and generates Verilog code to force a read cycle on all address locations. If the data at all locations (addresses) equals its previous value, then it is assumed that the data at memory locations has not been changed, otherwise a statement to flash a message in the report file is written.

Please amend Paragraph [237] as follows:

[237] Test 358 and Algorithm 359 ~~363~~ tests for port unknown. Whenever the task is invoked, it checks if the output port is unknown, else a statement to flash an error message in the report file is generated.

Please amend Paragraph [238] as follows:

[238] Test 360 and Algorithm 361 ~~364~~ is suitable for testing a port at high impedance. Whenever the task is invoked, it checks if the output port is set to "Z", else a statement to flash an error message in the report file is generated.

Please amend Paragraph [239] as follows:

[239] Test 362 and Algorithm 363 ~~365~~ tests for port no change. Whenever the task is invoked, it checks if the output port is showing the previous data, else a statement to flash an error message in the report file is generated.

Please amend Paragraph [240] as follows:

[240] Test 364 and Algorithm 300 ~~366~~ tests for corrupt data bit memory. In case of certain signal violations, some bits of the data are corrupted. To check for such cases, it is first needed to find what value is expected at the output port and a read cycle is forced at the specified memory address and finally the output at the port is compared with the calculated value. If the two values do not match, a statement to flash an error message is generated in the report file.

Please amend Paragraph [241] as follows:

[241] Test 366 and Algorithm 300 ~~366~~ tests for corrupt mask bit for memory. In case of certain signal violations, some bits of the mask are corrupted. To check for such cases, it is first needed to know what value is expected at the output port, then a read cycle is forced at the specified memory address and finally the output at the port is compared with the calculated value. If the two values do not match, a statement to flash an error message in the report file is written.

Please amend Paragraph [276] as follows:

[276] Particularly, at step 611 the port name is read, then the cycle named at step 612. If, at step 613, the FTYPE value is comprised between "1" and "3", then the Verilog code to generate calls and test cases

for FTYPE with one, two or three signals changing is accordingly generated at step 615 according to algorithm 616, otherwise an error occurs (step 614 ~~615~~).

Please amend Paragraph [291] as follows:

[291] At steps ~~step~~ 653 ~~to~~ 655 and 657 a check is made to verify whether the "TTYPE" order is comprised between "1" and "3", in which case Verilog tasks for TTYPE test cases of that order and corresponding calls are generated (step 654 [[-]] 656 and 658). Otherwise, an error is reported.